REMARKS

Claims 1-25 were presented for examination and all were rejected under 35 U.S.C. §§ 102 and 103, and also provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-32 of co-pending Application number 10/044394. Applicants are amending claims 1, 3, 5, 8, 11, 12, 14-16, 20, 21, and 25, canceling claims 2, 17, and 22, and adding claims 26-28.

In paragraph 1 of the Office Action, Applicants were reminded of the duty to fully disclose information under 37 CFR 1.56. Such reminder is very much appreciated.

In paragraph 3, it was indicated that the Specification had not been checked for all possible minor errors. To the best of Applicants' knowledge, the Specification does not include typographical errors, and showing errors with specificity would be very much appreciated.

In paragraph 4, claim 1 was objected to because of informalities. The recitation of "determining the access time" should be changed to "determining an access time." Claim 1 is being amended including the change. Withdrawal of the objection is respectfully requested.

In paragraph 5, claims 1-25 were provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unptentable over claims 1-32 of co-pending Application No. 10/044394. It was indicated that "[a]lthough the conflicting claims are not identical, they are not patentably distinct from each other because both memory systems comprise substantially the same elements of a system and method for managing a memory system . . . ." The claims in this application are being amended that include limitations distinguished from the

claims in the co-pending application, and therefore withdrawal of this rejection is solicited.

CLAIM REJECTIONS UNDER 35 U.S.C. § 102 – Hughes

In paragraphs 6 and 7 of the Office Action, claims 1-6, 8-13, 16-18, and 21-23 were rejected under 35 U.S.C. § 102 (b) as being anticipated by U.S. patent number 5,784,582 to Hughes ("Hughes"). The rejection is traversed. The Office Action failed to show every element of the claimed invention disclosed in Hughes.

The Office Action asserted "Hughes discloses a system and method for managing access latency by prioritizing memory access requests among a plurality of data paths based on configuration parameters, wherein the configuration parameters comprise location, size and direction of the transfer in combination with information of the current request." While this assertion is true, the claimed invention is not anticipated by the teaching of Hughes.

Regarding claims 1 and 16, the Office Action asserted "Hughes discloses a system and method substantially as claimed comprising the steps of: upon accessing the memory system for a piece of data used by a first process [i.e., request] determining the access time to acquire the piece of data in the memory system; comparing [i.e., selecting] the determined access time to a threshold [i.e., based on the parameter provided to the selection processor by the control state registers 109 (Fig. 3; column 5, lines 46-48)]; and taking actions based on the results of the comparing step [Fig. 4, column 2, lines 10-43]."

The Office Action failed to show that claim 1 is anticipated by Hughes. The Office Action failed to show complete correspondence between the claim limitations and the teaching of Hughes, and therefore failed to establish that the claim is anticipated. In general, the claimed invention is about comparing an access time to

acquire a piece of data to a threshold and taking actions based on results of such comparison. In contrast, Hughes is about prioritizing the requests for access to the shared memory system.

The Office Action failed to show "upon accessing the memory system for a piece of data used by a first process, determining an access time to acquire the piece of data in the memory system." While Hughes discloses "requests" for access to the shared memory, it does not teach "determining the access time to acquire the piece of data in the memory system."

The claimed "comparing the determined access time to a threshold" does not correspond to the cited "i.e., selecting" and "i.e., based on the parameters provided to the selection processor 108 by the control state registers 109." The Office Action failed to indicate what was being selected. The Office Action failed to show a threshold. The Office Action failed to show the "determined access time." Consequently, the Office Action also failed to show "comparing the determined access time to a threshold." Further, the Office Action failed to show "wherein a value of the threshold is selected based on one or a combination of cost of switching processes for execution, and whether the value is a realistic time for a memory access."

Hughes' cited Fig. 3 shows a block diagram of a shared memory arbiter/controller (col. 3, lines 42-43). Hughes' col. 5, lines 46-48 recite "[t]he request selection processor 108 is coupled to the control state registers 109 to provide parameters for the selection process."

Hughes' "selecting a request for access to the shared memory based on the parameter provided to the selection processor 108 by the control registers 109" does not correspond to the claimed "comparing the determined access time to a threshold" because, as can bee seen, "selecting a request for access to the shared memory" and

"comparing the determined access time to a threshold" have no patentably relationship to one another. Hughes' parameters have no bearing on the claimed access time (for the piece of data) or the threshold for that access time. Hughes' parameters are used for selecting and prioritizing requests and include location, size, and direction of the transfer in combination with information of the current request to reduce access latency. Hughes' parameters thus do not correspond to the claimed access time. The claimed invention is not about reducing access latency, but about taking actions based on the results of comparing the access time to a threshold.

Hughes' Fig. 4 is a flowchart illustrating the process executed in the shared memory arbiter (col. 3, lines 45-46) in which the controller receives multiple requests for access to the shared memory. If more than one request has the highest priority, then a process is executed to select the optimum request. However, if only one request has the highest priority, then the highest priority request is selected (col. 6, lines 63-66). Hughes' col. 2, lines 10-43 is part of the summary discussing "a shared memory architecture" (col. 2, lines 11-12) in which the shared memory generates requests for access to the shared memory. The requests have characteristics including a starting address, a length and an access type, which are processed in a memory controller. In one aspect, the memory controller includes logic responsive to configuration parameters to control the data paths sharing the memory. The configuration parameters comprise a data path priority parameter to manage the latency of stored requests from the plurality of data paths. Further, a data path can be given highest priority and be processed ahead of requests from other data paths, and thus provides protection to the high priority data path to ensure best case access to the shared memory. These cited paragraphs have no bearing on the claimed elements. Controlling the priority of data paths sharing the memory in Hughes has nothing to do with determining the access time to acquire the piece of data in the memory system,

or comparing the determined access time to a threshold, or taking actions based on results of the comparing step.

Additionally, Hughes does not disclose the limitation "[the actions] including postponing execution of the first process and allowing execution of a second process," which was part of claim 2 and is now in claim 1.

Claim 2 is being canceled, and part of its limitation is now in claim 1. Similarly, claim 17 is being canceled and part of its limitation is now in claim 16. The paragraph of col. 2, lines 31-56 of Hughes was recited in the rejection of claim 2 and claim 17. However, this paragraph discloses supplying a current request for access to the shared memory and selecting the next request with higher priority to be processed ahead of other request, which is patentably distinguished from, and has nothing to do with, the claimed "postponing execution of the first process and allowing execution of a second process."

Hughes does not disclose the following limitations now in claim 1:

"wherein a value of the threshold is selected based on one or a combination of

cost of switching processes for execution, and

whether the value is a realistic time for a memory access."

Regarding claim 3-4, 9 and 12, even though the cited figure 2 of Hughes discloses the SDRAM Arbiter/Controller Logic 72, it does not disclose that the SDRAM Arbiter/Controller Logic 72 perform the step of postponing execution of the first process and allowing execution of a second process upon a notification from a latency manager regarding a relationship between the determined access time and the threshold. The allegation that the latency manager is inherent is without merit. The mere fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish inherency. For the sake of argument that the latency is inherent "in order to determine the priority of the current request and to set the control

state registers 109 accordingly," as asserted in the Office Action, this assertion is patentably distinguished from the claimed "the latency manager determining the relationship [between the determined access time and the threshold] independent from the intelligence."

Regarding claims 3, 15, and 17, even though the limitation of the "latency manager notifying the intelligence that the determined access time is close to, equal to, or greater than the threshold; the latency manger performing the step of determining independent from the intelligence," is being canceled, it is for the record that this limitation is not disclosed in Hughes. The claimed invention is being amended to claim in the alternative language, not because that the canceled is taught by the prior art.

Regarding claim 4, Hughes does not disclose the claimed "the intelligence is selected from a group consisting of a processor working with the memory system, an operating system working with the memory system, software running on the processor, and a memory manager managing the memory system."

Regarding claim 5, the cited paragraphs of col. 5 lines 49-50 and col. 7, lines 57-61 do not disclose that "the actions further include monitoring the memory system or a system using the memory system." In fact, col. 5 lines 49-50 are about monitoring the requests in the shared memory pipeline. Col. 7 lines 57-61 are about monitoring the process in progress. The cited monitored requests and process are patentably distinguished from the claimed monitored memory system or system using the memory system.

Even though the limitation of "if the step of comparing indicates that the determined access time is close to, equal to, or greater than the threshold, further comprising the step of include monitoring the memory system or a system using the memory system" is being canceled, it is for the record that Hughes does not disclose

this limitation. The claimed invention is being amended to claim in the alternative language, not because the canceled limitation is taught by the prior art.

Regarding claims 6 and 18, the cited Fig. 4, the cited paragraphs of col. 5, line 63 through col. 6 line 5 and col. 6 lines 46-49 are not about the claimed selecting the determined access time as the longest access time of a plurality of access times each of which corresponds to a memory access in a multiple memory access. Even though col. 5 line 63 through col. 6 line 5 discloses selecting a next optimum request, this is not patentably equivalent to selecting the determined access time as the longest access time of a plurality of access times. The cited paragraph of col. 6 lines 46-49 is about selecting the request according to the parameters. However, the parameters include location, size, and direction of the transfer of the current access but do not include time such as access time or longest access time of a plurality of access times.

Regarding claim 8, the cited paragraph of col. 2, lines 44-56 does not disclose the claimed step of updating the previously determined access time to the determined access time if the determined access time is greater than the previously determined access time. In fact, that paragraph is about selecting requests to manage pipeline fullness.

Regarding claim 10, the cited paragraph of col. 2, lines 31-56, which was discussed above, is not about changing the determined access time upon performing a task selected from a group consisting of changing the threshold, initiating an interrupt to an intelligence working with the memory system, and postponing execution of the first process and allowing execution of a second process.

Regarding claim 11, the cited paragraph of col. 2, lines 44-56, which was discussed above, is not about the determined access time being selected from a time to access at least one subsystem.

Regarding claim 13, the cited figure 4 and the cited paragraphs are not about the data being accessed from a subsystem having a shorter access time to a subsystem having a longer accessed time or in a non-sequential order.

Regarding claims 21-23, the Office Action asserted that "Hughes teaches a computer implemented process" without providing evidence, and the assertion is therefore improper. The assertion that it is inherent that "the program accomplishing the procedures must be carried or stored on a computer medium" is also improper because the mere fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish inherency.

CLAIM REJECTIONS UNDER 35 U.S.C. § 102 – Lamberts

In paragraph 8, claims 1, 5-8, 11, 13, 16, 18-19, 21, and 23-24 were rejected under 35 U.S.C. § 102 (e) as being anticipated by U.S. patent number 6, 418,510 to Lamberts ("Lamberts").

The Office Action alleged that "Lamberts discloses a system and method for memory management that makes cache decisions based on a cost function wherein the cost function is calculated as a function of cache access time. Data with higher cost (i.e., higher access time) is added or kept in the cache while data with lower cost (i.e., lower access time) is not stored in the cache." While this assertion is true, the teaching of Lamberts does not anticipate the claimed invention.

Regarding claim 1 and 16, the Office Action cited FIG. 3, references 58 and 60; FIG. 4, references 90 and 92; FIG. 3 elements 62 and 66; FIG. 4, reference 94 and 98; FIG. 3, references 64, 68, 70, and 72; FIG. 4 references 96, 100, 102, and 104; col. 4 lines 29-55.

The cited FIG. 3, references 58 and 60 are about calculating a cost function for writing data block to disk; calculating the cost function for each data block currently stored in the cache. The cited FIG. 4, references 90 and 92 are about calculating the

cost function for not storing the data in cache and calculating the cost function for replacing each cached data block. The cited FIG. 3, elements 62 and 66 are about comparing the cost function for the new data block to the cost function for the cached data block. The cited FIG. 4, reference 94 and 96 are about comparing the cost function for the new data block to the cost function for the cached data block and not storing the new data block in cache if the cost function of the new data block has a lower cost function.

The cited FIG. 3, references 64, 68, 70, and 72 are about writing data block to disk if the data block has the lowest cost function, overwriting cached read data with new data block if the read command type has the lowest cost function and writing cached write data to disk and overwriting cached write data block with new data block if the write command type has the lowest cost function. The cited FIG. 4, references 96, 100, 102, and 104 are about not storing data block in cache if the data block has the lowest cost function overwriting cached read data with new data block if the read command type has the lowest cost function, writing cached write data to disk and overwriting cached write data with new data block if the write command type has the lowest cost function.

The cited col. 4 lines 29-55 disclose managing a cache based on the access time of the command under consideration wherein if a data block has a higher access time, it is preferably kept or added into cache while data with a lower access time is preferably not stored in cache.

As can be seen, these cited references do not disclose determining an access time to acquire the piece of data in the memory system. These cited references do not disclose a threshold, "wherein a value of the threshold is selected based on one or a combination of cost of switching processes for execution, and whether the value is a realistic time for a memory access." As a result, these cited paragraphs cannot

logically disclose comparing the determined access time to the threshold. Further, these references do not teach "taking actions . . ., including postponing execution of the first process and allowing execution of a second process."

Regarding claim 5, the Office Action cited FIG 3, element 62 and 66; FIG. 4, references 94 and 98; column 9 lines 10-40. FIG. 3, elements 62 and 66 and FIG. 4, references 94 and 98, as cited above, and col. 9 lines 10-40 do not disclose the canceled limitation if the step of comparing indicates that if the determined access time is close to, equal to, or greater than the threshold, further comprising the step of monitoring the memory system or a system using the memory system. These recitations do not disclose the current limitation of amended claim 5 which recites "wherein the actions include monitoring the memory system or a system using the memory system." The cited paragraph of col.9, lines 10-40 discloses calculation of Td as constrained by the system requirements and specifications. Td may be computed by calculating the estimated access time between each command in the queue and the new data command, and averaging the times. Alternatively, only commands whose distance in time from the new command is below a threshold value are combined to obtain the average. Alternatively, Td may be the estimated access time between the new data and the closest command in the command queue. Even though this paragraph discloses "threshold," but the threshold relates to the distance from the command; it does not relate to the threshold for a memory access. This cited paragraph also discloses the disk surface with concentric tracks, which has nothing to do with the claimed limitation. None of the citations discloses the claimed "monitoring the memory system or a system using the memory system."

Regarding claims 6 and 18, the Office Action also cited column 9, lines 10-40, which does not disclose the claimed that the determined access time is selected as the

longest access time of a plurality of access times each of which corresponds to a memory access in a multiple memory access.

Regarding claims 7, 19, and 24, the Office Action cited the cache buffer 42, disk drive 36, figure 2, and col. 9 lines 10-40. Again the paragraph of col. 9, lines 10-40 does not disclose the claimed limitation that if the piece of data is missed in the subsystem having the shorter access time, then the determined access time being that of the subsystem having the longer access time.

Regarding claim 8, the Office Action also cited col. 9, lines 10-40, which does not disclose the limitation of updating a previously determined access time to the determined access time if the determined access time is greater than the previously determined access time.

Regarding claim 11, the Office Action cited figures 3 and 4 and col. 4, lines 29-55, which, as cited above, do not disclose the claimed limitation that the determined access time is selected from a time to access at least one subsystem.

Regarding claim 13, the Office Action mistakenly mentioned Hughes, but probably meant Lamberts, col. 9, lines 10-40. However, this cited paragraph does not disclose the claimed limitation that the data is accessed from a subsystem having a shorter access time to a subsystem having a longer access time or in a non-sequential order.

Regarding claims 21 and 23, the Office Action again mentioned Hughes, but probably meant Lamberts. The Office Action asserted that "Hughes [sic, Lamberts] teaches a computer implemented process" without providing evidence, and the assertion is therefore improper. The assertion that it is inherent that "the program accomplishing the procedures must be carried or stored on a computer medium" is also improper because the mere fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish inherency.

CLAIM REJECTIONS UNDER 35 U.S.C. § 103 – Lamberts and Eickemeyer

In paragraphs 9 and 10, claims 2-4, 9-10, 12, 14-15, 17, 20, 22, and 25 were rejected under 35 U.S.C. § 103 (a) as being unpatentable over Lamberts and U.S. patent number 6,049,867 to Eickemeyer ("Eickemeyer"). The rejection is traversed. The claim invention is patentably distinguished from Lamberts and Eickemeyer, either alone or in combination. The alleged motivation for combining the two teachings is improper.

Regarding claims 2-4, 9-10, 12, 14-15, 17 and 20, the Office Action asserted "Lamberts teaches the method for managing a memory system substantially as claimed including the steps of: earmarking a subsystem from the plurality of subsystems [i.e., determining whether to overwrite data based on the estimated access time (see Figure 3 and Figure 4)]; and determining an order for data to be accessed from a subsystem having a shorter access time to a subsystem having a longer access time [i.e., Calculate cost, (Figure 3, references 58 and 60; Figure 4, references 90 and 92)]." The Office Action then admitted "[h]owever, Lamberts does not specifically teach an intelligence for postponing the current process and allowing a second process to execute." The Office Action continues "Eickemeyer teaches a method for memory management to reducing memory access latency utilizing a process or thread switch to allow the switching between multiple threads in response to the occurrence of an event that indicates long memory latency may occur. In an event of a cache miss, a first thread is suspended allowing a second thread to access the cache memory [Abstract, column 4, lines 27-55; and column 5, lines 4-7]. The Office Action then concluded "it would have been obvious for one skilled in the art at the time the invention was made to implement the system and method for managing a memory system as taught by Lamberts and incorporate Eickemeyer's teachings to include a

process switch to postpone of a current process and allow a second process to execute in the event of a cache miss. One skilled in the art would have been motivated to do so, because the utilization of a process switch provides further memory access latency reduction and eliminates the need for complex, replication of pipeline latches and pipeline states as pointed out by Eickemeyer on column 4, line 27 through 55."

The alleged motivation for combining the teaching of Lamberts and Eickemeyer is improper because it is merely a conclusory statement cited from Eickemeyer, and no evidence that suggests the combination was provided. While Eickemeyer indicated that "[b]y only switching threads in response to such events the necessity for increased complexity and replication of pipeline latches and additional pipeline states is avoided," there is no suggestion in either Lamberts or Eickemeyer that teaches the claimed invention as a whole that includes the limitation "determining an access time to acquire the piece of data in the memory system; comparing the determined access time to a threshold; and taking actions based on results of the comparing steps; including postponing execution of the first process and allowing execution of a second process. Further, even though Eickemeyer discusses termination of the first thread and selecting a second process for execution, it is in the context of cache miss in a level two or higher cache, not in the context of based on results of comparing the determined access time to a threshold wherein a value of the threshold is selected based on one or a combination of cost of switching processes for execution and whether the value is a realistic time for a memory access. Therefore, the alleged motivation is insufficient to support a prima facie case of obviousness.

Further, regarding claim 2-4, 9-10, 12, and 17, Eickemeyer does not provide the limitations in those claims that are missed in Lamberts. For example, Eickemeyer does disclose the limitation in claim 1, such as "determining an access time to acquire the piece of data in the memory system; comparing the determined access time to a

threshold; and taking actions based on results of the comparing step; . . . wherein a value of the threshold is selected based on one or a combination of cost of switching processes for execution, and whether the value is a realistic time for a memory access." Eickemeyer does not disclose limitations in each of the dependent claim, either.

Regarding claims 14, 20, and 25, as discussed above, Lamberts and Eickemeyer, either alone or in combination, do not disclose the claimed comparing an access time of a subsystem to a threshold; "a value of the threshold being selected based on one or a combination of cost of switching processes for execution, and whether the value is a realistic time for a memory access."

Lamberts' determining whether to overwrite data based on the estimated access time does not correspond to the claimed "earmarking a subsystem from a plurality of subsystem." The claimed subsystem is earmarked so that if the data is missed in this earmarked system, then execution of the first process is postponed for the second process to be executed, which is patentably distinguished from determining whether to overwrite data. The earmarked subsystem is also "based on results of the comparing step."

The cited calculate cost, figure 3, references 58 and 60; figure 4, references 90 and 92 do not correspond to the claimed "determining an order for data to be accessed from a subsystem having a shorter access time to a subsystem having a longer access time. As cited above, the cited FIG. 3, references 58 and 60 are about calculating a cost function for writing data block to disk; calculating the cost function for each data block currently stored in the cache. The cited FIG. 4, references 90 and 92 are about calculating the cost function for not storing the data in cache and calculating the cost function for replacing each cached data block. None of the teaching in these cited paragraphs relate to the claimed limitation.

Even though Eickemeyer discusses that "a first thread is suspended allowing a second thread to access the cache memory," it's not in the context of the above discussed limitations of the claimed invention, e.g., comparing an access time of a subsystem, earmarking the subsystem based on results of the comparing step, etc. Therefore, showing of a prima facie case for obviousness failed. The claimed invention is patentably distinguished from Lamberts and Eickemeyer, either alone or in combination.

The alleged motivation for combining Lamberts and Eickemeyer is improper. The Office Action asserted that "it would have been obvious for one skilled in the art . . . to implement the system and method for managing a memory system as taught by Lamberts and incorporate Eickemeyer's teachings to include a process switch to postpone of a current process and allow a second process to execute in an event of a cache miss. One skilled in the art would have been motivated to do so, because the utilization of a process switch provides further memory access latency reduction and eliminates the need for complex, replication of pipeline latches and pipeline states . . . ." This alleged motivation is a mere recitation of Eickemeyer without showing any suggestion for combining from either Lamberts or Eickemeyer. Therefore, a prima facie of obviousness failed.

Regarding claims 22 and 25, the Office Action again mentioned Hughes, but probably meant Lamberts. The Office Action asserted that "Hughes [sic, Lamberts] teaches a computer implemented process" without providing evidence, and the assertion is therefore improper. The assertion that it is inherent that "the program accomplishing the procedures must be carried or stored on a computer medium" is also improper because the mere fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish inherency.

ADDED CLAIMS

Claims 25-28 are being added and depend from claim 21. Therefore, claims 25-28 are patentable for at least the same reasons as claim 21. Claim 25-28 are also patentable for their additional limitations corresponding to claims 5, 8, and 11 because claims 25-28 recite limitations corresponding to claims 5, 8, and 11, respectively. Limitations in claims 25-28 are supported in the Specification, and therefore no new matter is added.

SUMMARY

In conclusion, Applicants respectfully submit that pending and added claims clearly present subject matter that is patentable over the prior art of record, and therefore request that the Examiner withdraw the rejections of the pending claims, consider the added claims, and pass the application to issue.

Respectfully submitted,

Date: 6/21/04 By: _____

Tuan V. Ngo, Reg. No. 44,259
IP Administration
Legal Department, M/S 35
Hewlett-Packard Company
P. O. Box 272400
Fort Collins, CO 80527-2400
Phone (408) 447-8133
Fax (408) 447-0854